

**Course Name** : PYTHON PROJECTS MASTERY: FROM NOVICE TO EXPERT  
**Duration** : 3 Days (Physical Classroom / Virtual Live Instructor)  
**Skill Level** : Beginner

**COURSE DESCRIPTION:**

Embark on a transformative journey through "Python Projects Mastery: From Novice to Expert," where you'll delve into the world of Python programming through practical, hands-on projects. Designed for beginners and aspiring developers, this course takes you from foundational Python concepts to advanced project implementations. You'll start by mastering essential Python syntax, data structures, and functions, setting a solid groundwork for your coding endeavors. As you progress, you'll tackle a diverse array of projects spanning natural language processing, application development, game creation, web applications, and more.

Each project is meticulously crafted to reinforce your learning and build real-world programming skills. Whether you're creating a chatbot, developing a web application, or designing interactive games, you'll gain invaluable experience and confidence in Python programming. By the end of this course, you'll emerge as a proficient Python developer, equipped with the knowledge and expertise to tackle complex projects and solve real-world challenges with Python.

**WHAT WILL YOU LEARN?**

In "Python Projects Mastery: From Novice to Expert," you will learn to master Python programming through a series of hands-on projects designed to elevate your skills from beginner to advanced levels. You will gain proficiency in fundamental Python concepts such as syntax, data structures, functions, and object-oriented programming principles. Through practical applications in natural language processing, application development, game creation, web development, and more, you will acquire the ability to solve real-world problems using Python. By the end of the course, you will have the confidence and expertise to independently develop robust Python projects and explore further opportunities in software development and data analysis.

**PREREQUISITE:**

This course is suitable for beginners as it covers Python basics, making it accessible even if you have no prior programming experience.

**METHODOLOGY:**

This program will be conducted with interactive lectures, PowerPoint presentations, discussions, and practical exercises. This course can be conducted as instructor-led (ILT) or virtual instructor-led training (VILT).

**JOB SCOPE:**

Upon completion of this course, candidates may pursue the following career paths:

- Software Engineer/Developer
- Data Scientist
- AI/Machine Learning Engineer
- Data Engineer
- DevOps Engineer

## MODULE 1: PYTHON BASICS

- Welcome
- Installing Python and IDEs (e.g., PyCharm, VS Code, Jupyter Notebook)
- Introduction to Python Shell and Script Mode
- Writing and running Python scripts
- Basic syntax: variables, data types, and operators
- Control structures: if statements, loops (for, while)
- Defining and calling functions
- Importing and using modules
- Introduction to standard libraries
- Lists, tuples, sets, and dictionaries
- Basic operations and methods for each data structure
- Reading from and writing to files
- Working with different file types (txt, CSV)

## APPLICATION DEVELOPMENT

### PROJECT 1: ALARM APP

- Introduction to building an alarm app
- Setting up alarms with Python

### PROJECT 2: ATM APP

- Designing an ATM interface
- Implementing basic ATM functionalities

### PROJECT 3: CALCULATOR APP

- Building a simple calculator
- Adding basic arithmetic operations

### PROJECT 4: CLOCK APP

- Creating a digital clock
- Updating the time in real-time

### PROJECT 5: COMPRESSION APP

- Introduction to data compression
- Building a file compression app

### PROJECT 6: CURRENCY CONVERSION APP

- Using APIs for currency conversion
- Building the conversion interface

### PROJECT 7: NOTIFICATION APP

- Introduction to desktop notifications
- Implementing notification features in Python

#### **PROJECT 8: TODO APP**

- Creating a task management app
- Adding, deleting, and updating tasks

#### **PROJECT 9: SCREEN BRIGHTNESS APP**

- Adjusting screen brightness with Python
- Building a simple interface for control

#### **PROJECT 10: BUILD TASK MANAGER APP**

- Creating a system task manager
- Monitoring and managing system tasks

#### **PROJECT 11: LAPTOP BATTERY CHECK APP**

- Checking battery status with Python
- Building a battery monitoring app

### **GAME DEVELOPMENT**

#### **PROJECT 12: DICE GAME**

- Introduction to game development with Python
- Coding a simple dice rolling game

#### **PROJECT 13: GUESS THE NUMBER GAME**

- Building a number guessing game
- Adding user interaction and feedback

#### **PROJECT 14: ROCK PAPER SCISSOR GAME**

- Coding the classic Rock Paper Scissors game
- Implementing game logic and user input

### **WEB DEVELOPMENT**

#### **PROJECT 15: FIRST WEB APP**

- Introduction to web applications with Python
- Building your first web app

#### **PROJECT 16: GET VALUE FROM FORM FIELDS**

- Creating web forms
- Retrieving and processing user input

#### **PROJECT 17: LOGIN PAGE WITH PYTHON AND MYSQL**

- Setting up MySQL for user authentication
- Building a login interface

## **NATURAL LANGUAGE PROCESSING(NLP)**

### **PROJECT 18: JARVIS APP**

- Introduction to Jarvis App
- Setting up the environment
- Coding the main features

### **PROJECT 19: CHATGPT WITH PYTHON**

- Overview of ChatGPT
- Integrating ChatGPT with Python
- Implementing a basic ChatGPT interface

### **PROJECT 20: PYTHON VOICE ASSISTANCE APP**

- Introduction to voice assistance
- Libraries needed for voice recognition
- Building the voice assistant step-by-step

### **PROJECT 21: PYTHON SIMPLE CHATBOT**

- Quick setup of a simple chatbot
- Basic chatbot functionality
- Extending chatbot features

### **PROJECT 22: EXTRACT DATA FROM PDF**

- Tools for PDF data extraction
- Writing code to extract text and data from PDF files

### **PROJECT 23: TEXT SENTIMENT ANALYSIS**

- Understanding sentiment analysis
- Tools and libraries for sentiment analysis
- Implementing sentiment analysis in Python

### **PROJECT 24: SPELL CHECK**

- Introduction to spell checking
- Libraries for spell checking in Python
- Writing code for spell check functionality

### **PROJECT 25: SIMILARITY ANALYSIS**

- Concept of text similarity
- Techniques for similarity analysis
- Coding similarity analysis in Python

## **PROJECT 26: LANGUAGE DETECTION**

- Introduction to language detection
- Tools for detecting language
- Implementing language detection in Python

## **PROJECT 27: WORD COUNT**

- Basics of word count
- Writing a Python script for word counting

## **PROJECT 28: YOUTUBE VIDEO DOWNLOADER**

- Introduction to YouTube Video Downloader
- Implement YouTube Video Downloader

## **PROJECT 29: AUTOMATED FILE BACKUP**

- Introduction to file Backup
- Implement Automated File backup

## **PROJECT 30: WEB SCRAPING USING PYTHON**

- Basics of web scraping
- Tools and libraries for web scraping
- Implementing web scraping to extract data

## **CONCLUSION**

- QA
- Useful References and Books
- Feedback